

This listing of claims will replace all prior versions and listings of claims in the application:

Listing of Claims

1. (Currently Amended) A system of software components, which can be invoked during runtime by an application program running under an operating system to display a parent_object and a child object, and wherein the child object is displayed in accordance with the following background color inheritance order:

if a background color of the child object is declared, the child object is displayed with the declared background color;

if the background color of the child object is not declared, and a background color is globally defined for the system of components, the child object is displayed with the globally defined background color, which is independent of the operating system; and

if the background color of the child object is not declared, and the background color setting is not globally defined for the system of components, the child object is displayed with the background color of the parent_object.

2. (Currently Amended) The system as recited in claim 1, wherein the ~~object is parent~~ and child objects are part of a graphical user interface associated with the application program.

3. (Original) The system as recited in claim 1, wherein the application program is written in Java programming language.

4. (Currently Amended) The system as recited in claim 1, wherein the system of software components is comprises various software components within the Swing application program interface (API).

5. (Original) The system as recited in claim 1, wherein the operating system comprises a standard computer operating system such as Windows, Unix or OS/2.

6. (Currently Amended) The system as recited in claim 1, wherein the child object is one of multiple objects within a layout associated with the application program.

7. (Canceled)

8. (Currently Amended) A method for color inheritance between a parent object and a child object, which are displayed ~~using by~~ a system of software components ~~within~~ created by an application program running on an operating system, wherein the method comprising is adapted to display the child object in accordance with the following background color inheritance order:

if a background color of the child object is declared, displaying the child object with the declared background color;

if the background color of the child object is not declared, and a background color is globally defined for the system of components, displaying the child object with the globally defined background color, which is independent of the operating system; and

A⁴
if the background color of the child object is not declared, and the background color setting is not globally defined for the system of components, displaying the child object with the background color of the parent.

9. (Currently Amended) The method as recited in claim 8, wherein the ~~object is parent and child objects~~ are part of a graphical user interface associated with the application program.

10. (Original) The method as recited in claim 8, wherein the application program is written in Java programming language.

11. (Currently Amended) The method as recited in claim 8, wherein the system of software components is ~~comprises various software components within the~~ Swing application program interface (API).

12. (Original) The method as recited in claim 8, wherein the operating system comprises a Windows, Unix or OS/2 computer operating system.

13. (Currently Amended) The method as recited in claim 8, wherein the child object is one of multiple objects within a layout associated with the application program.

14. (Canceled)

✓ 15. (Currently Amended) A computer-readable storage device, comprising:

a windows-based operating system;

a system of software components within the operating system;

A4 | a parent object and a child object, both created at runtime by an application running under the operating system, wherein the child object is displayed by the system of software components in accordance with the following background color inheritance order;

if a background color of the child object is declared when it is created, the child object is displayed with the declared background color;

if the background color of the child object is not declared when it is created, and a background color is globally defined for the system of components, the child object is displayed with the globally defined background color, which is independent of the operating system; and

if the background color of the child object is not declared when it is created, and the background color setting is not globally defined for the system of components, the child object is displayed with the background color of the parent.

16. (New) The system as recited in claim 1, wherein the parent object is an AWT-based object or a Swing-based object.

A5 | 17. (New) The system as recited in claim 1, wherein the system of software components comprises a lightweight peer component and a lightweight proxy component for displaying the child object with either the globally defined or inherited background color, if the background color of the child object is not declared.

A⁵
18. (New) The system as recited in claim 17, wherein the lightweight peer component is adapted to redirect a method call for getting the background color to the lightweight proxy component.

19. (New) The system as recited in claim 18, wherein the lightweight proxy component is adapted to translate the method call to determine whether the child object should be displayed using the globally defined or inherited background color.
